



RAPIDS Simulator

Rapids Simulator



What is RAPIDS?

- An event-driven simulator for distributed real-time systems
- Simulator itself also distributed, runs on a heterogeneous cluster of workstations

Issues in Real-Time Systems

- Timeliness
- Recovery from failure
- Fault-tolerance



Why RAPIDS?

- Assist in the development and testing of recovery algorithms
- Analyze other hardware and software components of real-time systems



Simulator Structure

- Physical machines
- PVM message passing interface
- PVM processes
- Simulated system



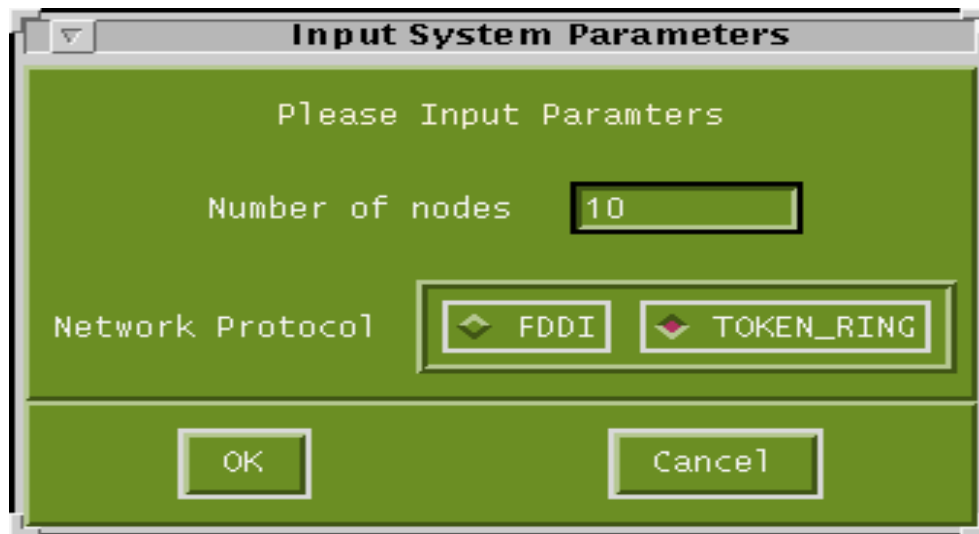
Simulated System Components

- Console - The link between the GUI and the simulator
- Clock - Ensures causality, uses the breathing time buckets technique
- Network - Simulates connection network between nodes.
- Node(s)



Node Components

- Tasks: Synthetic and real
- Scheduling algorithms
- Allocation algorithms
- Checkpointing
- Recovery actions



A screenshot of a software dialog box titled "Input System Parameters". The dialog has a green background and a grey title bar. Inside, the text "Please Input Parameters" is centered. Below it, the label "Number of nodes" is followed by a text input field containing the value "10". Further down, the label "Network Protocol" is followed by two radio button options: "FDDI" (which is selected) and "TOKEN_RING". At the bottom of the dialog are two buttons: "OK" and "Cancel".

Input System Parameters

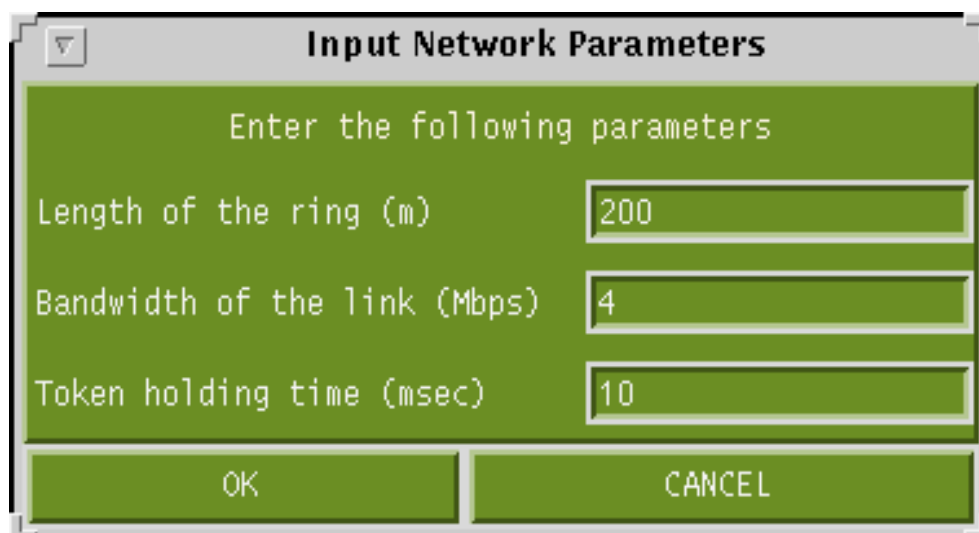
Please Input Parameters

Number of nodes 10

Network Protocol ☒ FDDI ☐ TOKEN_RING

OK Cancel

Choice of network protocol and number of nodes



A screenshot of a software dialog box titled "Input Network Parameters". The dialog has a green background and a grey title bar. Inside, the text "Enter the following parameters" is centered. Below it are three labels with corresponding text input fields: "Length of the ring (m)" with the value "200", "Bandwidth of the link (Mbps)" with the value "4", and "Token holding time (msec)" with the value "10". At the bottom of the dialog are two buttons: "OK" and "CANCEL".

Input Network Parameters

Enter the following parameters

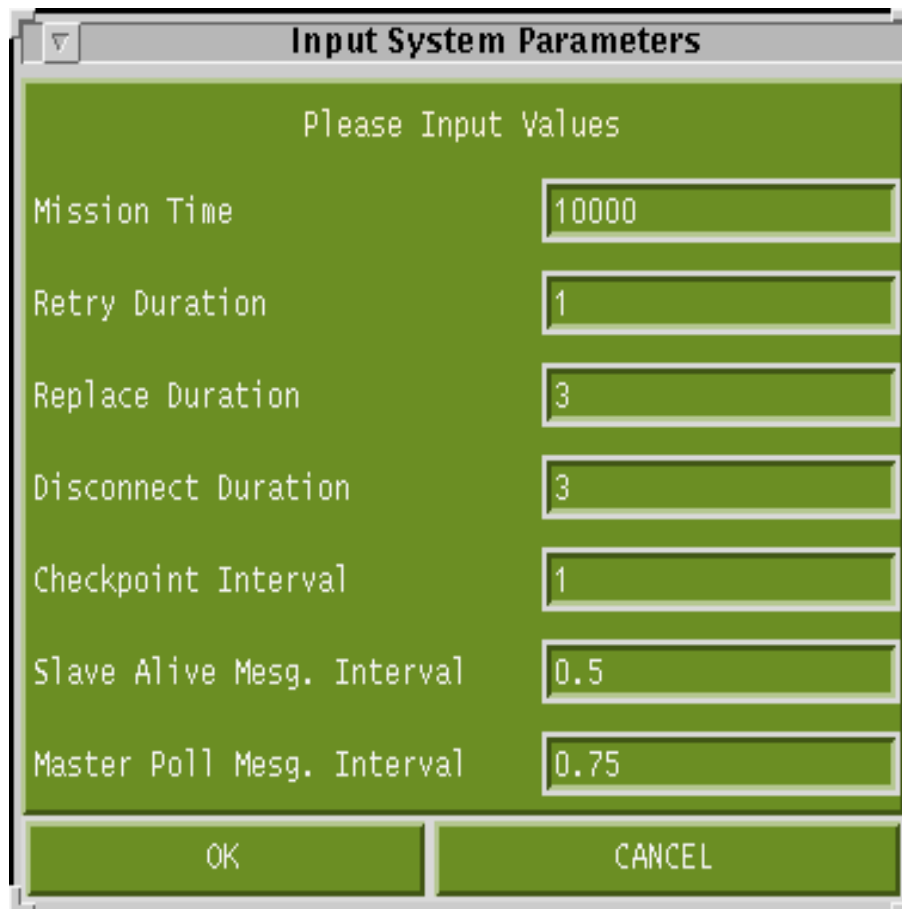
Length of the ring (m) 200

Bandwidth of the link (Mbps) 4

Token holding time (msec) 10

OK CANCEL

Choice of network parameters



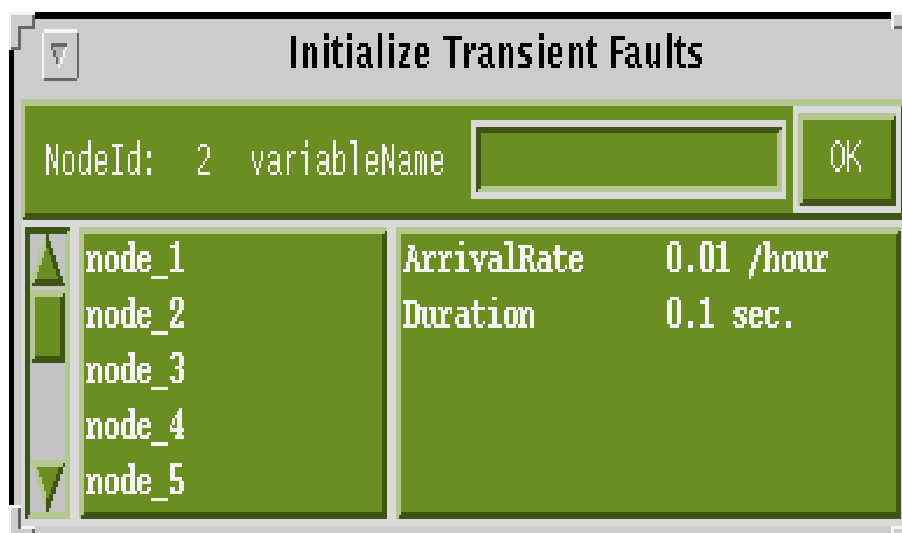
Input System Parameters

Please Input Values

Mission Time	10000
Retry Duration	1
Replace Duration	3
Disconnect Duration	3
Checkpoint Interval	1
Slave Alive Mesg. Interval	0.5
Master Poll Mesg. Interval	0.75

OK CANCEL

Choice of various system and recovery parameters



Initialize Transient Faults

NodeId: 2 variableName OK

node_1	ArrivalRate	0.01 /hour
node_2	Duration	0.1 sec.
node_3		
node_4		
node_5		

Choice of fault rates

Task Editor

TASK ID :
SUBTASK ID :

1 NUM_TASKS 1

2 variableName

taskId 1

TASK_TYPE
PERIOD_INTERVAL
NUM_PERIODS
NUM_REDUNDANCY
NEXT_RELEASE_TIME
NUM_SUBTASKS

3
10
-1
1
0
5

subtaskId 1
subtaskId 2
subtaskId 3
subtaskId 4
subtaskId 5

NODEID
OFFSET
EXECTIME
DEADLINE
PRIORITY
CHECKPOINT_SIZE
NUM_RECVMESGS
NUM_SENDMESGS

-1
0
2
10
0
20
0
0

LOAD TASK SET

ADD NEW TASK

SAVE TASK SET

ADD TASK FROM FILE

OK

CANCEL



Recovery Algorithms

- Static, in which the user specifies ahead of time what fixed recovery action is to be taken
- Our RAMP algorithm, a dynamic offline algorithm



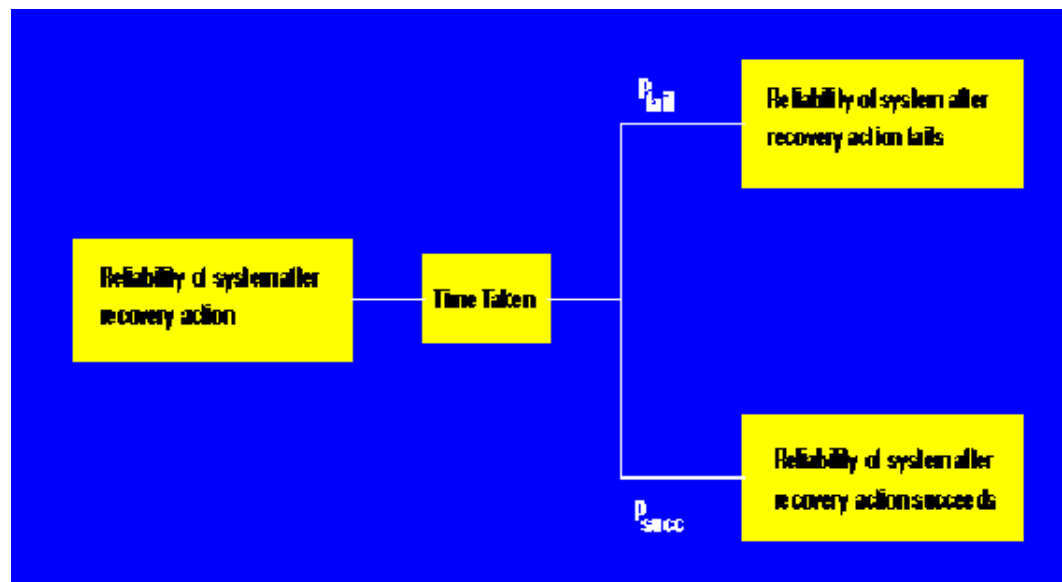
Recovery Actions

- Retry the task from its last checkpoint
- Replace the faulty node by a spare (if available)
- Disconnect the faulty node and redistribute its tasks over the remaining nodes



RAMP Algorithm

- RAMP generates efficient recovery policies for a given system depending on:
 - ★ Operating environment
 - ★ Workload
 - ★ System failure characteristics
 - ★ Remaining mission time





Why a Benchmark?

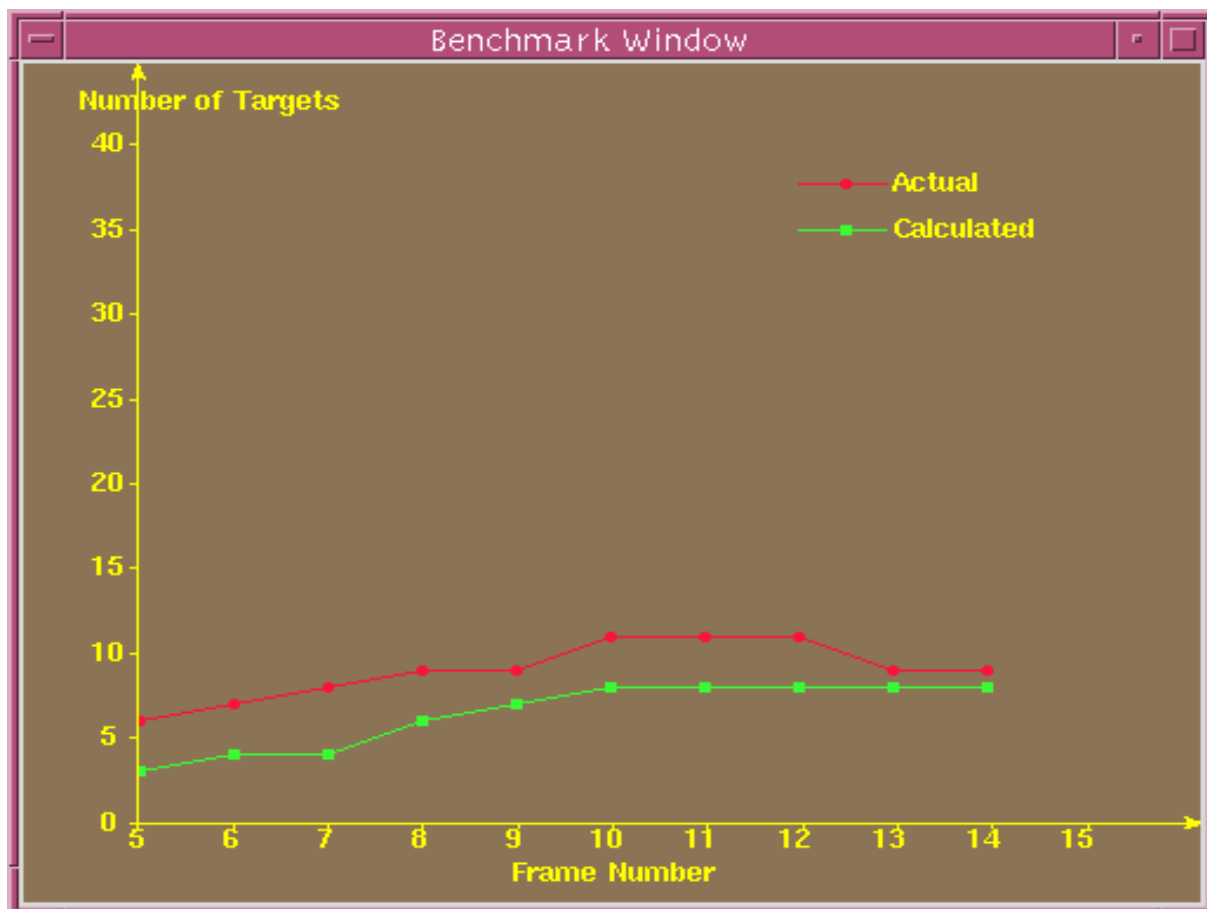
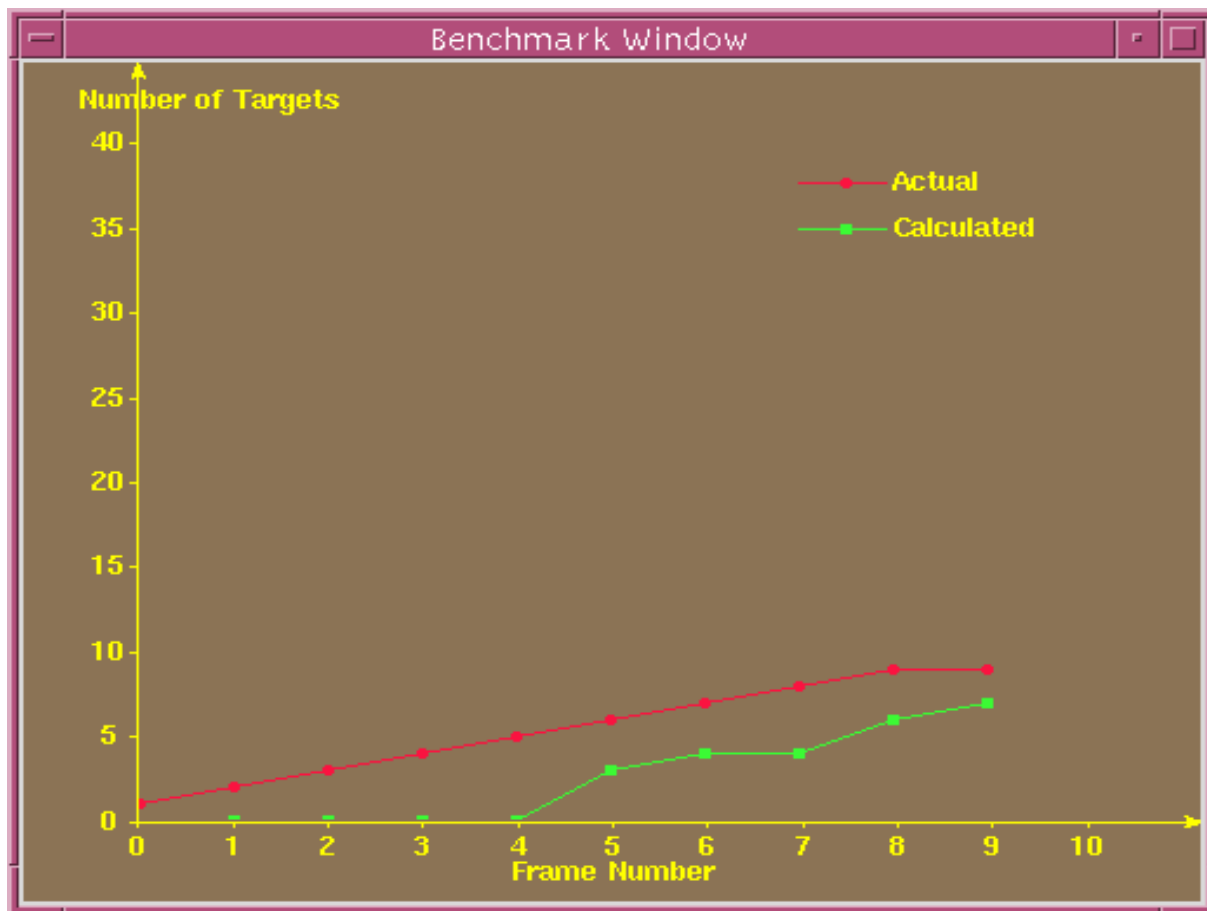
- Analyze the performance impact of the implemented algorithms and policies

Multi-Hypothesis Tracking Benchmark

- A real-time target tracking benchmark of the DARPA benchmark suite

Implementing the Benchmark in RAPIDS

- The benchmark tasks are executed and results, in terms of targets successfully tracked, are returned





Results

- **RAPIDS: A distributed simulator testbed for reliable real-time systems**
- **Helps designers compare different policies and algorithms for reliable real-time systems design**
- **Modular and easily extendable for new network protocols, allocation and scheduling algorithms and recovery techniques**
- **By exposing the potential bottlenecks, can help fine-tune the design of the architecture and operating system for a given real-time system**



- A preliminary version of the simulator has already been transferred to JPL and Cal Tech.
- We plan to have a student spend several months at JPL to tailor the simulator to their needs.